

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-131195

(43)Date of publication of application : 13.05.1994

(51)Int.Cl.

G06F 9/45

(21)Application number : 04-282714

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 21.10.1992

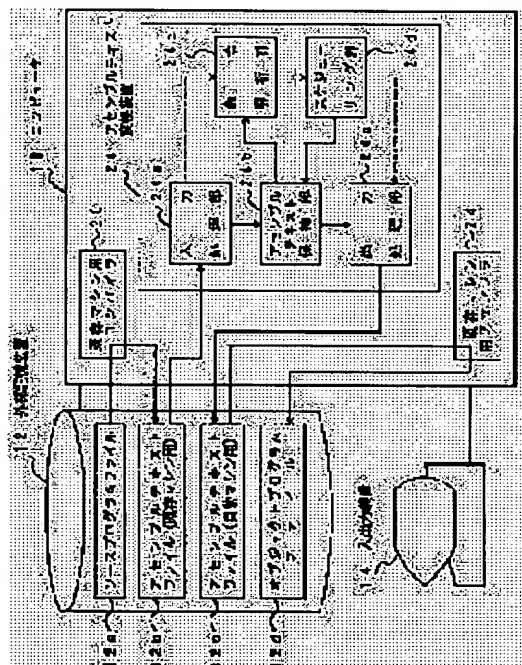
(72)Inventor : YAMADA TOMOHIKO
MIYAMOTO IZURU
YOSHINO YASUAKI

(54) ASSEMBLY TEXT CONVERTER

(57)Abstract:

PURPOSE: To make the high-speed execution of an instruction string possible on a target machine.

CONSTITUTION: This converter is provided with an instruction analysis part 24c for analyzing the dependency relation of respective instructions within the instruction string expressed by an assembly text prepared for an existing machine and a scheduling part 24d for rearranging the instructions of the instruction string corresponding to the instruction execution conditions of the target machine capable of parallelly executing the plural instructions depending on the arrangement of the instructions provided with interchangeability between the existing machine and an instruction system based on an analyzed result by the instruction analysis part 24c. Then, the assembly text prepared for the existing machine is rearranged into the instruction string executable at a high speed by the target machine.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

(11)特許出願公開番号

(43)公開日 平成6年(1994)5月13日

技術表示箇所

320 D

(74)代理人 弁理士 鈴江 武彦

- 1 -

【特許請求の範囲】

【請求項1】 既存マシン用に作成されたアセンブルテキストが表現する命令列中の、各命令の依存関係を解析する命令解析手段と、

前記命令解析手段による解析結果に基づいて、前記既存マシンと命令体系に互換性がある命令の並びによっては複数命令を並列実行可能な目的マシンの命令実行条件に応じて、前記命令列の命令の並べ変えを行なうスケジューリング手段と、

を具備し、

既存マシン用に作成されたアセンブルテキストを、目的マシンで高速実行可能となる命令列に並べ変えることを特徴とするアセンブルテキスト変換装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ソースプログラムをもとに生成されたアセンブルテキストを変換するアセンブルテキスト変換装置に関する。

【0002】

【従来の技術】近年、高性能RISCマイクロプロセッサの出現により、ワークステーションは、急激に性能を向上させている。一般に、RISCプロセッサは、動作周波数を上げることによって性能向上を図っている。

【0003】従って、既存のRISCマイクロプロセッサ（以下、既存のRISCマイクロプロセッサを備えたコンピュータを既存マシンと称する）と、性能向上が図られたRISCマイクロプロセッサ（以下、性能向上が図られたRISCマイクロプロセッサを備えたコンピュータを目的マシンと称する）とは命令体系が同一となっている。

【0004】このような既存マシンと命令体系が同一である目的マシンに対しては、ソースプログラムから既存マシン用のコンパイラ及びアセンブラを使用してオブジェクトプログラムを生成している。

【0005】ところで、性能向上の目的で動作周波数を上げることは、一方では高速信号の取扱いや発熱などの実装上での問題、あるいは半導体設計の困難さなどの問題が発生してしまう。

【0006】このようなことに対処するために、動作周波数を上げることの他に、命令の並びによってはプロセッサ内部で複数命令を並列実行可能にすることで、性能を向上させることが考えられる。

【0007】しかしながら、このような手段によって性能を向上させたとしても、従来のように、既存マシン用のコンパイラ及びアセンブラを使用してオブジェクトプログラムを用いていたのでは、目的マシンにおける並列実行の実行性能を十分に発揮させることができない。

【0008】

【発明が解決しようとする課題】このように従来では性能向上が図られた目的マシンに対して、既存マシン用の

コンパイラ及びアセンブラを使用してオブジェクトプログラムが生成されていたために、複数命令を並列実行可能にした目的マシンの並列実行性等の実行性能を十分に発揮させることができなかった。

【0009】本発明は前記のような点に鑑みてなされたもので、目的マシン上での命令列の高速実行を可能にするアセンブルテキスト変換装置を提供することを目的とする。

【0010】

10 【課題を解決するための手段】本発明は、既存マシン用に作成されたアセンブルテキストが表現する命令列中の、各命令の依存関係を解析する命令解析手段と、前記命令解析手段による解析結果に基づいて、前記既存マシンと命令体系に互換性がある命令の並びによっては複数命令を並列実行可能な目的マシンの命令実行条件に応じて、前記命令列の命令の並べ変えを行なうスケジューリング手段とを具備し、既存マシン用に作成されたアセンブルテキストを、目的マシンで高速実行可能となる命令列に並べ変えることを特徴とする。

【0011】

20 【作用】このような構成によれば、既存マシン用に作成されたアセンブルテキスト、すなわち既存マシン用コンパイラによって作成されたアセンブルテキストを変換することにより、複数命令を並列実行可能な目的マシンにおいて実行されるオブジェクトプログラムが生成される。この際、アセンブルテキストが表現する命令間の依存関係をもとに、目的マシンの命令実行条件に従って、すなわち目的マシンのアーキテクチャを考慮して、命令の並べ変えが行なわれるので、目的マシンの実行動作性能を有効に利用した高速実行可能な目的マシン用プログラムが得られる。

【0012】

30 【実施例】以下、図面を参照して本発明の一実施例を説明する。図1は本実施例に係わる情報処理システムの構成を示すブロック図である。図1に示すように、情報処理システムは、コンピュータ10、外部記憶装置12、及び入出力装置14が設けられている。

【0013】コンピュータ10は、既存のRISCマイクロプロセッサを備えたコンピュータ（既存マシン）であって、性能向上が図られたRISCマイクロプロセッサを備えたコンピュータ（目的マシン）用のオブジェクトプログラムを生成する。目的マシン（図示せず）のプロセッサは、例えばスーパースカラ・プロセッサであり、既存マシンと命令体系に互換性があり、命令の並びによってはプロセッサ内で複数命令を同時に実行可能となっている。

40 【0014】外部記憶装置12は、各種ファイルを格納するためのものである。状況に応じて、ソースプログラムファイル12a、アセンブルテキストファイル12b、12c、オブジェクトプログラムファイル12d等

3

が格納される。入出力装置14は、コンピュータ10とエンドユーザとのインターフェイスであり、アセンブルテキスト変換の要求等を行なうために操作される。

【0015】コンピュータ10には、既存マシン用コンパイラ20、既存マシン用アセンブラ22、及びアセンブルテキスト変換装置24が設けられる。既存マシン用コンパイラ20は、外部記憶装置12内のソースプログラムファイル12aを読み込み、既存マシン用のアセンブルテキストを生成し、アセンブルテキストファイル12bとして外部記憶装置12に格納する。アセンブルテキスト変換装置24は、既存マシン用のアセンブルテキストを、目的マシンでの高速実行に適したアセンブルテキストに変換し、アセンブルテキストファイル12cとして外部記憶装置12に格納する。既存マシン用アセンブラ22は、アセンブルテキストファイル12cを読み込み、オブジェクトプログラムを生成し、オブジェクトプログラムファイル12dとして外部記憶装置12に格納する。

【0016】アセンブルテキスト変換装置24は、入力処理部24a、アセンブルテキスト保持部24b、命令解析部24c、スケジューリング部24d、及び出力処理部24eが設けられている。

【0017】入力処理部24aは、既存マシン用コンパイラ20によって生成された、既存マシン用のアセンブルテキストファイル12bを読み込み、命令解析部24c及びスケジューリング部24dが処理するための内部形式に変換し、アセンブルテキスト保持部24bに保持させる。アセンブルテキスト保持部24bは、内部形式に変換されたアセンブルテキストを保持するためのものである。

【0018】命令解析部24cは、アセンブルテキスト保持部24b内の内部形式のアセンブルテキストが表わす命令列をもとに命令依存グラフを作成することによって命令間の依存関係を解析する。

【0019】スケジューリング部24dは、命令解析部24cによる命令解析に基づいて、命令列の並べ換えを行ない、目的マシンにおける並列実行の実行性能を十分に発揮させることが可能な命令列を表わすアセンブルテキストに変換する。

【0020】出力処理部24eは、スケジューリング部24dによって命令列の並べ換えが行なわれた内部形式のアセンブルテキストを元のテキスト形式に変換し、アセンブルテキストファイル12cとして出力する。次に、本実施例のアセンブルテキスト変換装置24の動作について、図2に示すフローチャートを参照しながら説明する。

【0021】なお、外部記憶装置12内には、既存マシン用コンパイラ20によって生成されたアセンブルテキストが、アセンブルテキストファイル12bとして格納されているものとする。

4

【0022】ここで、コンピュータ10に対して、入出力装置14を介してユーザからのアセンブルテキストの変換要求がなされると、アセンブルテキスト変換装置24が起動される。

【0023】ユーザによって変換対象となるファイルが指定されると、アセンブルテキスト変換装置24の入力処理部24aは、該当するアセンブルテキストファイル12bを外部記憶装置12から読み込む。入力処理部24aは、読み込んだアセンブルテキストを内部形式に変換して、アセンブルテキスト保持部24bに保持させる(ステップS1)。

【0024】アセンブルテキスト保持部24bに変換対象とするアセンブルテキスト(内部形式による)が保持されると、命令解析部24cは、アセンブルテキスト内の各命令の依存関係を解析し、各命令をノードとした有向線分で表わした命令依存グラフを作成する(ステップS2)。

【0025】ここで、具体例を用いて説明する。図3はアセンブルテキスト変換前の命令列の一例を示している。なお、目的マシンのアーキテクチャによって命令実行は次のように条件(命令実行条件)づけられるものとする。

- a. 3命令まで同時実行可能とする。
- b. 1d(ロード)結果は、同時に使用できないものとする。
- c. 四則演算結果は、1d命令、st(ストア)命令、演算命令で同時に使用可能とする。
- d. 演算命令は、同時に2つまで実行可能とする。
- e. 1d命令が第3命令の場合は、次のサイクルで使えないものとする。

命令解析部24cは、図3に示す命令列から、図4に示すような、命令依存グラフを作成する。

【0026】スケジューリング部24dは、命令解析部24cにより作成された命令依存グラフの構造を変えない範囲内で、目的マシンにおいて並列実行命令が増加するように命令を並べ変える(ステップS3)。

【0027】スケジューリング部24dは、例えば図5に示すように命令列の並べ換えを行なったものとする。並べ換えが行なわれた命令列を表わすアセンブルテキスト(内部形式)は、アセンブルテキスト保持部24bに保持される。

【0028】出力処理部24eは、アセンブルテキスト保持部24bに保持された、内部形式のアセンブルテキストを元のテキスト形式に変換し、アセンブルテキストファイル12cとして外部記憶装置12に格納する。

【0029】このアセンブルテキストファイル12cは、既存マシン用アセンブラ24によってオブジェクトプログラムに変換され、オブジェクトプログラムファイルとして外部記憶装置12に格納される。ここで、図3

5

に示す並べ変え前の命令列の動作クロック数と、並べ変え後の命令列の動作クロック数とを比較する。

【0030】まず、並べ変えの前の命令列について判別する。命令(3)は、前述した命令実行条件bにより、命令(2)のld結果を同時に使えないので、命令(1)(2)が同時に実行される。次に、演算命令である3つの命令(3)(4)(5)が連続しているので(sft(シフト)命令は演算命令とする)、命令(3)(4)が同時に実行される。命令(5)(6)(7)は、命令実行条件a、cにより同時に実行可能である。ここで、命令(7)はld命令であり、第3命令であるので、命令実行条件eによりインタロックが発生する。演算命令である2つの命令(8)(9)は、1クロック待った後、同時に実行される。命令(10)(11)は、命令実行条件cにより、同時に実行される。従って、並べ変えが行なわれない場合の、図3に示す命令列の実行は、6クロック要する。

【0031】一方、アセンブルテキスト変換装置24によって並べ変えが行なわれた命令列は、図4に示す命令依存グラフの構造を変えず、インタロックが生じないように並べ変えられている。すなわち、命令(1)(2)(6)、命令(7)(3)(5)、命令(4)(8)、命令(9)(10)(11)が、それぞれ同時に実行される。従って、並べ変えが行なわれた場合の、図5に示す命令列の実行は、4クロック要する。

【0032】このようにして、命令依存グラフに基づき、この構造を変えない範囲で、目的マシンのアーキテクチャを考慮して、並列実行命令が増加するように、そしてインタロックができる限り生じないようにして、命令の並べ変えを行なうので、目的マシンにおける並列実行の実行性能を十分に発揮させることができる。

【0033】また、アセンブルテキストの変換によって目的マシンに適したオブジェクトプログラムを生成することになるので、既存マシン用コンパイラ20、既存マ

6

シン用アセンブラ22を変更することなく、既存マシンから目的マシンへの変更に対処することができる。なお、前記実施例では既存マシンと目的マシンは別のものとして説明しているが、同一のものであっても良い。また、RISCマイクロプロセッサを備えたコンピュータとしているが、プロセッサを特に限定するものではない。

【0034】

【発明の効果】以上のように本発明によれば、既存マシン用に生成されたアセンブルテキストを、既存マシンと命令体系に互換性があり、命令の並びによってはプロセッサ内で複数命令の同時実行が可能な目的マシンに適するように、アセンブルテキストが表わす命令列の意味を変えない範囲で適切に命令の並べ変えを行なうので、目的マシン上での命令列の高速実行が可能となるものである。

【図面の簡単な説明】

【図1】本発明の一実施例に係わる情報処理システムの構成を示すブロック図。

【図2】アセンブルテキスト変換装置24の動作を説明するためのフローチャート。

【図3】アセンブルテキスト変換前の命令列の一例を示す図。

【図4】図3に示す命令列に基づく命令依存グラフの一例を示す図。

【図5】アセンブルテキスト変換後の命令列の一例を示す図。

【符号の説明】

10…コンピュータ、12…外部記憶装置、14…入力装置、20…既存マシン用コンパイラ、22…既存マシン用アセンブラ、24…アセンブルテキスト変換装置、24a…入力処理部、24b…アセンブルテキスト保持部、24c…命令解析部、24d…スケジューリング部、24e…出力処理部。

10 コンピュータ

12 外部記憶装置

14 入出力装置

20 既存マシン用コンパイラ

24 アセンブルテキスト変換装置

24a 入力部

24b アセンブルテキスト保持部

24c 命令解析部

24d スケジューリング部

24e 出力部

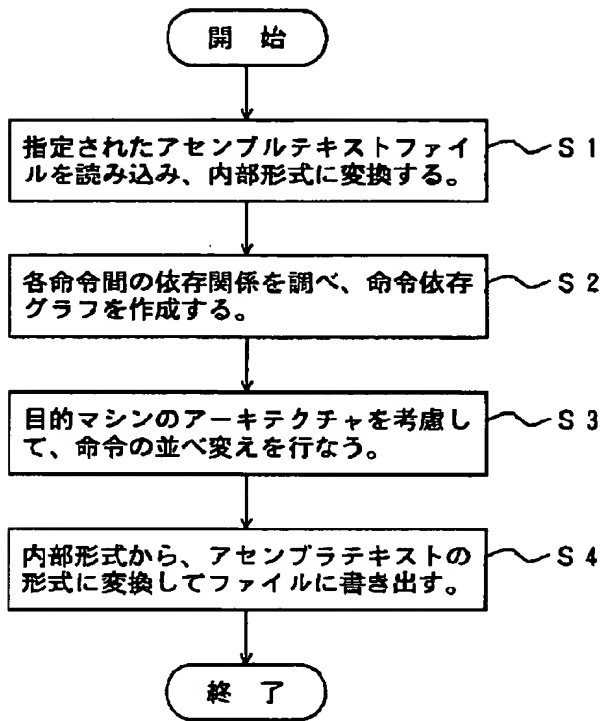
12a ソースプログラムファイル

12b アセンブルテキストファイル(既存マシン用)

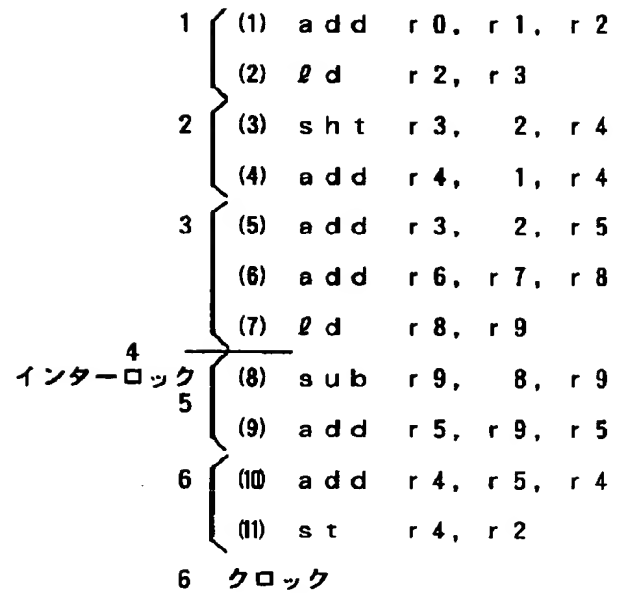
12c アセンブルテキストファイル(目的マシン用)

12d オブジェクトプログラムファイル

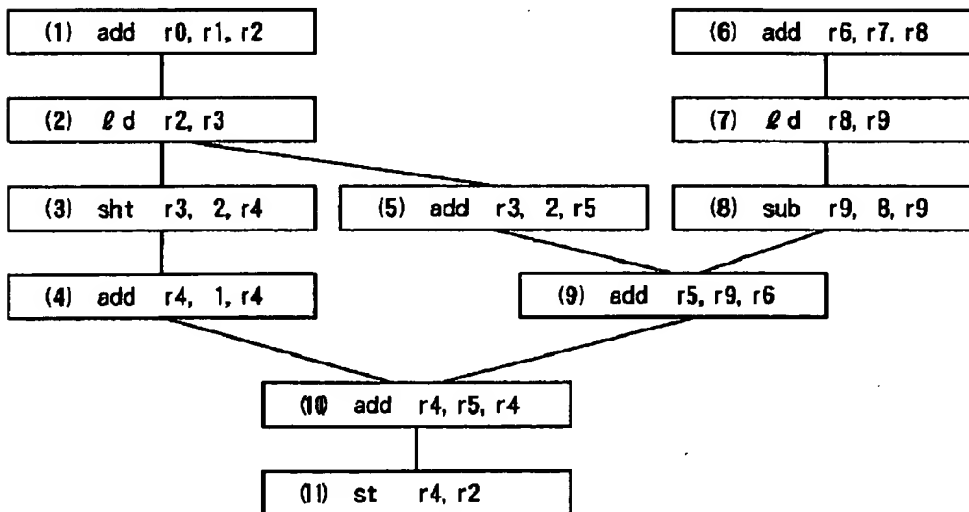
【図2】



【図3】



【図4】



【図5】

1	{	(1) add r0, r1, r2
		(2) ld r2, r3
		(6) add r6, r7, r8
2	{	(7) ld r8, r9
		(3) shl r3, 2, r4
		(5) add r3, 2, r5
3	{	(4) add r4, 1, r4
		(8) sub r9, 8, r9
4	{	(9) add r5, r9, r5
		(10) add r4, r5, r4
		(11) st r4, r2

4 クロック